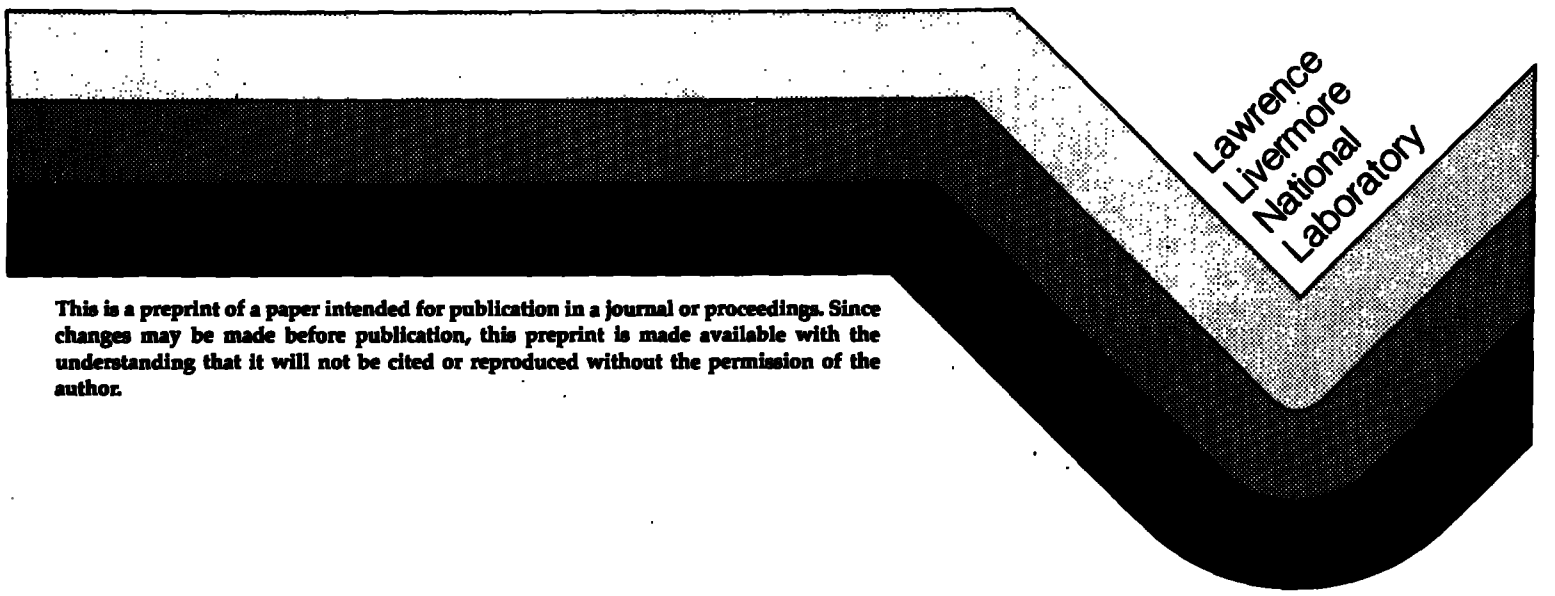
UCRL-92907
PREPRINT

SPLINE
(LLNL Micro-CAE Software)

Ray Pierce

This paper was prepared for submittal to
Micro-CAE Technology Transfer Workshop
Flagstaff, AZ - August 18-23, 1985

June 1985



The computer program SPLINE is an interactive, standalone program that calculates the natural cubic interpolatory spline fit to the data supplied by the user. The code presented and discussed briefly here is a modified version of the subroutine SPCOEF and function SPLINE programs found in reference 1.

The data input by the user is stored in two arrays, X1 and F1. The X1 array contains the array of X-values while the array F1 contains the corresponding function values. The X1 values must be distinct. The spline is determined in subroutine SPCOEF and evaluated in subroutine SPLINE. SPCOEF arranges the nodes in increasing order and stores this order in the array I1. The array X1 itself is not altered. SPCOEF then calculates the array S of second derivatives needed to define the spline.

SPLINE uses "natural end conditions" at the endpoints. That is, at the first and last data points the second derivative of the cubic spline is set equal to zero.

The user is given two data input options:

1. From a disk file that was generated by a previous run of SPLINE
2. Input by the user at SPLINE run time.

After the data has been input, the user has the opportunity to

1. Correct (change) individual data values
2. Add up to 10 more data points
3. Delete data points

When the user is finished correcting, adding, or deleting data, he/she is given the option of saving the data in a disk file.

The program then calculates the cubic interpolatory spline for the given data. SPLINE then asks the user if he/she wants the program to calculate the coefficients of the cubic curve for each sub-interval. The coefficients correspond to the following form of a cubic equation:

$$F(X)=A*X^3+B*X^2+C*X+D$$

If the user has SPLINE calculate the cubic coefficients, he/she is given the option of saving them on disk for later use.

The user then has the opportunity to input X-values for which he/she wants SPLINE to calculate F(X) values.

Finally, the user is asked if he/she wants SPLINE to go through the entire procedure for a new set of data.

VARIABLE LIST AND DEFINITIONS

A	Length of right side of sub-interval that the X-value of interest lies within.
ASCI	Used when analyzing user response for possible input error. Represents the ASCII code of response.
A$	User response to questions asked by the program. Also used as filename for disk input/output operations.
B	Length of left side of sub-interval that the X-value of interest lies within.
B$	Used to check user response in disk I/O operations; to determine if disk drive address has been included in the user input filename.
CC(I,J)	Cubic coefficients for each sub-interval.
D	Used in intermediate calculation for solving for the second derivatives.
DEC	Counter for the number of decimal points in the user input. Used to check for error in user input.
D$	Used to indicate which disk drive to use for disk I/O.
EPN	Counter for the number of exponents in the user input. Used to check for error in the user input.
FD	First derivative of the cubic spline curve at the point of interest.
FD(I)	The first derivatives of the spline at the data points.
FM$	Output formatting variable.
FT(I)	Temporary storage for the array F1(I)-used when deleting data.
F1(I)	F(X) values of input data.
H1	The length of the first (left) sub-interval.
H2	The length of the last (right) sub-interval.
H3	The length of the interior sub-interval that the X-value of interest is in.
H4,H5	Length of the sub-intervals used in calculating intermediate variables used to compute the second derivatives.
H9	Used in calculating F(X) for interior sub-interval.
I	General purpose counter.
II	General purpose counter.
IK	General purpose counter.
INUMN	Lower limit to possible user numerical input. Used in checking for input error.
INUMX	Upper limit to possible user numerical input. Used in checking for input error.
INUM$	User response to questions asked by the program. Used to check the validity of responses.
IT	Used in "DELETE" subroutine. Represents the subscripts of lines not deleted but moved to temporary storage.
I1	Subscript of the X1 array that contains the smallest value of the X1 array.
I1(I)	Index array that stores the subscripts of the X1 array such that X1(I1(1))<X1(I1(2))<X1(I1(3))... etc. This allows the data to be put in any X-order.
I2	Subscript of the X1 array that contains the second smallest value of the X1 array.

I3	Used to determine which interior interval the X-value of interest is located in.
I4,I5	Temporary storage of index array variables.
I6,I7,I8	Counters used to arrange the nodes in increasing X-order.
I9	Subscript of the X1 array element that contains the largest value of the X1 array.
J	General purpose counter.
K1	Temporary storage used in arranging the nodes in increasing X-order.
K2,K3	Temporary storage used in an intermediate calculation before solving for the second derivatives.
K4	Counter used in calculating the second derivatives.
L	General purpose counter.
LTR$	Used when checking user response for validity.
M$	Indicates if data is from disk (="I") or to disk (="O").
N	Number of valid data points.
NA	Number of additional data points (added by user).
ND	Number of data points deleted by the user.
ND(I)	Indicates which data points are to be deleted.
NE	Total number of data points that can be added during a run of SPLINE.
NF	Used for data input routine (initial input and subsequent input). Indicates the subscript of the first item of data to be input (=1 for initial input ; <> 1 for subsequent data input).
NGS	Used to count the number of negative signs (-) in the user response to questions.
NL	Used for data input. Indicates the subscript of the last item of data to be input.
NO	Indicates which option the user selected from a menu.
NP	The total number of data points to be used to calculate the spline-including the additional ones added by the user at a subsequent run of SPLINE.
NZ	Used to "page" the output to the screen.
N1	Equals N-1
N2	Equals N-2
N8	Used to "page" the output to the screen.
N9	Indicates the number of the data line (point) to be changed by the user (CHANGE routine).
PLS	Used to count the number of positive signs (+) in the user response to questions.
R1(I)	RHO array. An array of intermediate answers used to calculate the second derivatives.
S(I)	Array of second derivatives calculated by SPLINE.
S9	Equals F(X) at desired X.
T1	Results of intermediate calculation when solving for the second derivatives.
T1(I)	TAU array. An array of intermediate answers used to calculate the second derivatives.
X	The X-value that the user wants to find F(X) for.
XT(I)	Temporary storage of the X1 array; used in the DELETE routine.
X1(I)	X-values of the user input data.

REFERENCES

1. Shampine, Lawrence F., and Richard C. Allen, Jr., "Numerical Computing: an introduction," W.B. Saunders Company, Philadelphia, Pa., 1973.

```
 10 REM SPLINE
 20 REM This is a modified version of the
 30 REM subroutine SPCOEF and function SPLINE programs
 40 REM published on pp 234-238 of the book
 50 REM Numerical Computing : an introduction, by
 60 REM L.F. Shampine and R.C. Allen, (1973).
 70 REM This is a standalone program written in
 80 REM CP/M MBASIC
 90 FM$="##.###^^^^"
100 GOSUB 2920
110 GOSUB 2940
120 PRINT SPC(28) " S P L I N E "
130 GOSUB 2940
140 REM instructions follow
150 INPUT "Do you want instructions? (Y or N)",A$
160 IF A$="Y" OR A$="y" THEN 190
170 IF A$="N" OR A$="n" THEN 730
180 GOTO 150
190 GOSUB 2920
200 PRINT "SPLINE calculates the natural cubic interpolatory spline fit to"
210 PRINT "the data specified by the array of N nodes X1, with corresponding"
220 PRINT "function values in the array F1.  The nodes X1 must be distinct."
230 PRINT "The spline is determined in subroutine SPCOEF and evaluated in"
240 PRINT "Function SPLINE. SPCOEF arranges the nodes in increasing order and"
250 PRINT "stores this order in the array I1.  The array X1 itself is not"
260 PRINT "altered. SPCOEF then calculates the array S of second derivatives"
270 PRINT "needed to define the spline."
280 PRINT
290 PRINT "SPLINE uses 'natural end conditions' at the endpoints.  That is,"
300 PRINT "at the first and last data points the second derivative of the"
310 PRINT "interpolatory curve is set equal to zero."
320 PRINT
330 PRINT "The user is given two data input options:"
340 PRINT
350 PRINT "    1. from a disk file that was generated by a previous run"
360 PRINT "       of SPLINE"
370 PRINT "    2. input 'by hand' at SPLINE run time"
380 PRINT:PRINT:GOSUB 2970:GOSUB 2930
390 PRINT "After the data has been input, the user has an opportunity to:"
400 PRINT
410 PRINT "    1. correct (change) individual data points"
420 PRINT "    2. add up to 10 more data points"
430 PRINT "    3. delete data points"
440 PRINT
450 PRINT "When the user is finished correcting, adding, or deleting data"
460 PRINT "he/she is given the option of saving the data in a disk file."
470 PRINT
480 PRINT "The program then calculates the cubic interpolatory spline for"
490 PRINT "the given data."
500 PRINT
510 PRINT "SPLINE then asks the user if he/she wants the program to calculate"
520 PRINT "coefficients correspond to the following form of a cubic equation:"
530 PRINT
540 PRINT "                  F(X)= A*X^3+B*X^2+C*X+D"
550 PRINT
560 PRINT "If the user has SPLINE calculate the cubic coefficients, he/she"
570 PRINT "is given the option of saving them on disk for later use."
580 GOSUB 2970:GOSUB 2930
590 PRINT "The user then has the opportunity to input X-values for which"
600 PRINT "he/she wants spline to calculate the F(X) values."
```

```
610 PRINT
620 PRINT "Finally, the user is asked if he/she wants SPLINE to go through"
630 PRINT "the entire procedure for a new set of data."
640 PRINT
650 PRINT "This concludes the instructions."
660 GOSUB 2950:GOSUB 2970
670 GOTO 730
680 REM subroutine to dimension arrays
690 NE=10:NP=N+NE:N1=N-1
700 DIM X1(NP),F1(NP),S(NP),I1(NP),R1(NP),T1(NP),FD(NP),CC(NP,4)
710 DIM ND(NP),XT(NP),FT(NP)
720 RETURN
730 GOSUB 2920
740 GOSUB 2940
750 REM data input option: disk or by hand
760 INPUT "Data from disk (Y or N)";A$
770 IF A$="N" OR A$="n" THEN GOSUB 2920:GOSUB 2940:GOTO 870
780 IF A$="Y" OR A$="y" GOTO 810
790 GOTO 760
800 REM input data from disk
810 M$="I"
820 GOSUB 3790
830 INPUT#1,N:GOSUB 690
840 FOR I=1 TO N:INPUT#1,X1(I),F1(I):NEXT I
850 CLOSE:GOTO 920
860 REM enter data by hand
870 INPUT "How many data pairs are to be entered";INUM$
880 INUMX=250:INUMN=1:GOSUB 3270:IF INUM$="E" GOTO 870
890 N=INUM
900 GOSUB 690
910 NF=1:NL=N:GOSUB 2920:GOSUB 3010
920 GOSUB 2920
930 REM echo input data for check
940 PRINT "Input data echo check"
950 PRINT "Line number","  X1(N)  ","  F1(N)  "
960 N8=20:NZ=N
970 IF NZ>N8 GOTO 990
980 N8=NZ
990 FOR I=1 TO N8:PRINT I,:PRINT USING FM$;X1(I),F1(I):NEXT I:GOSUB 2980
1000 NZ=NZ-N8:IF NZ>0 GOTO 970
1010 GOSUB 2920:GOSUB 2940
1020 REM correct/add/delete data
1030 PRINT "Do you want to make corrections, additions,"
1040 INPUT "or deletions (Y or N)";A$
1050 IF A$="Y" OR A$="y" GOTO 1080
1060 IF A$="N" OR A$="n" GOTO 1540
1070 GOTO 1010
1080 GOSUB 2920:PRINT SPC(20) "Options":PRINT
1090 PRINT SPC(18) "1. Corrections"
1100 PRINT SPC(18) "2. Additions"
1110 PRINT SPC(18) "3. Deletions"
1120 GOSUB 2940:INPUT "Enter the number of the desired option";INUM$
1130 INUMX=3:INUMN=1:GOSUB 3270:IF INUM$="E" GOTO 1080
1140 NO=INUM:GOSUB 2920:GOSUB 2940
1150 ON NO GOTO 1170,1280,1330
1160 REM change/correct data
1170 INPUT "Input the line number of the data to be corrected";INUM$
1180 INUMX=N:INUMN=1:GOSUB 3270:IF INUM$="E" GOTO 1170
1190 N9=INUM
1200 PRINT "Current X1(";N9;") = ";X1(N9),"Current F1(";N9;") = ";F1(N9)
```

```
1210 PRINT:PRINT "Input new X1(";N9;") = ";:INPUT X1(N9)
1220 PRINT "Input new F1(";N9;") = ";:INPUT F1(N9)
1230 GOSUB 2920:GOSUB 2940:INPUT "Any more corrections (Y or N)";A$
1240 IF A$="N" OR A$="n" GOTO 920
1250 IF A$="Y" OR A$="y" GOTO 1170
1260 GOTO 1230
1270 REM add data
1280 INPUT "How many data pairs to be added? (max. of 10)";INUM$
1290 INUMX=NE:INUMN=0:GOSUB 3270:IF INUM$="E" GOTO 1280
1300 NA=INUM:NF=N+1:NL=N+NA:N=NL:N1=N-1
1310 GOSUB 3010:GOTO 920
1320 REM delete data
1330 INPUT "How many lines to be deleted";INUM$
1340 INUMX=N:INUMN=0:GOSUB 3270:IF INUM$="E" GOTO 1330
1350 ND=INUM:IF ND=0 GOTO 1010
1360 GOSUB 2920
1370 PRINT "Begin deleting ";ND; " lines. ":GOSUB 2940
1380 FOR I=1 TO ND
1390 PRINT "Line number to be deleted = ";:INPUT INUM$
1400 INUMX=N:INUMN=1:GOSUB 3270:IF INUM$="E" GOTO 1390
1410 X1(INUM)=999.999
1420 NEXT I
1430 IT=1
1440 FOR I=1 TO N
1450 XT(I)=0!:FT(I)=0!
1460 IF X1(I)=999.999 GOTO 1490
1470 XT(IT)=X1(I):FT(IT)=F1(I)
1480 IT=IT+1
1490 NEXT I
1500 FOR I=1 TO N
1510 X1(I)=XT(I):F1(I)=FT(I)
1520 NEXT I
1530 N=N-ND:N1=N-1:GOTO 920
1540 GOSUB 2920:GOSUB 2940
1550 REM save input data on disk?
1560 INPUT "Want to save input data on disk (Y or N)";A$
1570 IF A$="N" OR A$="n" THEN GOSUB 2920:GOSUB 2940:GOTO 1650
1580 IF A$="Y" OR A$="y" GOTO 1600
1590 GOTO 1560
1600 M$="O"
1610 GOSUB 3790
1620 PRINT#1,N:FOR I=1 TO N:PRINT#1,X1(I);F1(I):NEXT I
1630 CLOSE:GOSUB 2920:GOSUB 2940
1640 REM calculate spline
1650 PRINT SPC(25) "Calculating Spline"
1660 GOSUB 2940
1670 REM ******************
1680 REM spcoef calculates spline fit to data
1690 REM ******************
1700 REM arrange the nodes in increasing order. store the order in array i1.
1710 FOR I=1 TO N
1720 I1(I)=I
1730 NEXT I
1740 FOR I=1 TO N1
1750 I6=I+1
1760 FOR J=I6 TO N
1770 I7=I1(I)
1780 I8=I1(J)
1790 IF X1(I7)<=X1(I8) THEN 1830
1800 K1=I1(I)
```

```
1810 I1(I)=I1(J)
1820 I1(J)=K1
1830 NEXT J
1840 NEXT I
1850 N2=N-2
1860 REM calculate the elements of the arrays rho and tau.
1870 R1(2)=0
1880 T1(2)=0
1890 FOR I=2 TO N1
1900 K2=I1(I-1)
1910 I7=I1(I)
1920 K3=I1(I+1)
1930 H4=X1(I7)-X1(K2)
1940 H5=X1(K3)-X1(I7)
1950 T1=(H4/H5)*(R1(I)+2)+2
1960 R1(I+1)=-1/T1
1970 D=6*((F1(K3)-F1(I7))/H5-(F1(I7)-F1(K2))/H4)/H5
1980 T1(I+1)=(D-H4*T1(I)/H5)/T1
1990 NEXT I
2000 REM compute the array of second derivatives (s) for the natural spline.
2010 REM
2020 S(1)=0
2030 S(N)=0
2040 FOR I=1 TO N2
2050 K4=N-I
2060 S(K4)=R1(K4+1)*S(K4+1)+T1(K4+1)
2070 NEXT I
2080 GOSUB 2920:GOSUB 2940
2090 REM calculate cubic coefficients?
2100 INPUT "Do you want the cubic coefficients (Y or N)";A$
2110 IF A$="Y" OR A$="y" THEN GOSUB 2920:GOSUB 3080:GOTO 2140
2120 IF A$="N" OR A$="n" THEN GOSUB 2920:GOSUB 2940:GOTO 2270
2130 GOTO 2100
2140 REM save the cubic coefficients on disk
2150 PRINT:PRINT:PRINT
2160 INPUT "Do you want to save the cubic coefficients on disk (Y or N)";A$
2170 IF A$="Y" OR A$="y" GOTO 2200
2180 IF A$="N" OR A$="n" GOTO 2260
2190 GOTO 2160
2200 M$="O"
2210 GOSUB 3790
2220 PRINT#1,N:FOR I=1 TO N:PRINT#1,X1(I):NEXT I
2230 FOR I=1 TO N-1:FOR J=1 TO 4:PRINT#1,CC(I,J):NEXT J:NEXT I
2240 CLOSE
2250 REM interpolate at desired values of x
2260 GOSUB 2920:GOSUB 2940
2270 INPUT "Do you want to calculate F(X) for input values of X (Y or N)";A$
2280 IF A$="N" OR A$="n" THEN GOSUB 2920:GOSUB 2940:GOTO 2370
2290 IF A$="Y" OR A$="y" GOTO 2310
2300 GOSUB 2940:GOTO 2270
2310 INPUT "How many values of X to be input";INUM$
2320 INUMX=50:INUMN=1:GOSUB 3270:IF INUM$="E" GOTO 2310
2330 FOR IK=1 TO INUM:INPUT "Input X-value = ";X
2340 GOSUB 2440:PRINT:PRINT "F(X) = ";S9:PRINT:NEXT IK
2350 GOSUB 2970:GOSUB 2920:GOSUB 2940:GOTO 2270
2360 REM another spline fit to new data?
2370 INPUT "Want to do another spline fit with new data (Y or N)";A$
2380 IF A$="N" OR A$="n" GOTO 2410
2390 IF A$="Y" OR A$="y" THEN CLOSE:CLEAR:GOTO 90
2400 GOTO 2370
```

```
2410 GOSUB 2920:GOSUB 2940
2420 PRINT "This concludes the SPLINE program.":END
2430 REM ******************
2440 REM subroutine spline
2450 REM ******************
2460 REM This subroutine accepts as input the quantities N, X1, F1, S,
2470 REM and I1 as defined above and a number X at which the spline is
2480 REM to be evaluated.  This subroutine must be called once for each
2490 REM argument at which the spline is to be evaluated.
2500 REM
2510 REM If X<X1(I1(I1(1)), approximate the function by a straight line which
2520 REM passes through the point (X1(I1(1),F(I1(1))) and whose slope agrees
2530 REM with the slope of the spline at that point.  Similarly for
2540 REM X>X1(I1(N)).  These are called the "natural end conditions".
2550 REM
2560 REM
2570 REM
2580 I1=I1(1)
2590 IF X>=X1(I1) THEN 2690
2600 I2=I1(2)
2610 H1=X1(I2)-X1(I1)
2620 S9=F1(I1)+(X-X1(I1))*((F1(I2)-F1(I1))/H1-H1*S(2)/6)
2630 FD=(F1(I2)-F1(I1))/H1-H1*S(2)/6
2640 RETURN
2650 REM if X>X1(I1(N)) approximate the function by the straight line which
2660 REM passes through the point (X1(I1(N)), F1(I1(N))) and whose slope agree
2670 REM with the slope of the spline at that point.
2680 REM
2690 I9=I1(N)
2700 IF X<=X1(I9) THEN 2770
2710 I2=I1(N-1)
2720 H2=X1(I9)-X1(I2)
2730 S9=F1(I9)+(X-X1(I9))*((F1(I9)-F1(I2))/H2+H2*S(N-1)/6)
2740 FD=(F1(I9)-F1(I2))/H2+H2*S(N-1)/6
2750 RETURN
2760 REM for X1(I1(I1(1))<=X<=X1(I1(N)) calculate the spline fit.
2770 FOR I=2 TO N
2780 I3=I1(I)
2790 IF X<=X1(I3) THEN 2810
2800 NEXT I
2810 L=I-1
2820 I4=I1(L)
2830 I5=I1(L+1)
2840 A=X1(I5)-X
2850 B=X-X1(I4)
2860 H3=X1(I5)-X1(I4)
2870 H9=1/(2*H3)
2880 S9=A*S(L)*(A*A/H3-H3)/6+B*S(L+1)*(B*B/H3-H3)/6
2890 S9=S9+(A*F1(I4)+B*F1(I5))/H3
2900 FD=-S(L)*H9*A*A+S(L+1)*H9*B*B+(F1(I5)-F1(I4))/H3-(H3/6)*(S(L+1)-S(L))
2910 RETURN
2920 REM clear screen
2930 PRINT CHR$(12):RETURN
2940 REM print spacing subroutine
2950 FOR I=1 TO 10: PRINT: NEXT I
2960 RETURN
2970 REM 'wait' subroutine
2980 PRINT:PRINT "Hit any key to continue"
2990 A$=INKEY$:IF A$="" GOTO 2990
3000 RETURN
```

```
3010 REM subroutine to input data
3020 PRINT "Begin entering data pairs as: X-value, F(X)-value"
3030 PRINT "such as:  2.5,5.3"
3040 FOR I=NF TO NL
3050 PRINT "N = ";I;"      ";
3060 INPUT "X1(N),F1(N):";X1(I),F1(I)
3070 NEXT I:RETURN
3080 REM subroutine to solve for cubic coefficients
3090 REM need first derivitives at data points
3100 PRINT
3110 PRINT "INT                A              B              C              D"
3120 PRINT
3130 FOR II=1 TO N
3140 X=X1(II)
3150 GOSUB 2580
3160 FD(II)=FD
3170 NEXT II
3180 REM now solve for coefficients
3190 FOR I=1 TO N-1
3200 CC(I,1)=(S(I)-S(I+1))/(6*(X1(I)-X1(I+1)))
3210 CC(I,2)=(S(I)+S(I+1)-6*CC(I,1)*(X1(I)+X1(I+1)))/4
3220 CC(I,3)=FD(I)-3*CC(I,1)*X1(I)^2-2*CC(I,2)*X1(I)
3230 CC(I,4)=F1(I)-CC(I,1)*X1(I)^3-CC(I,2)*X1(I)^2-CC(I,3)*X1(I)
3240 PRINT I,CC(I,1),CC(I,2),CC(I,3),CC(I,4)
3250 NEXT I
3260 RETURN
3270 REM routine to check number input both for expected limits and to
3280 REM ascertain that no erronious letters or other characters are included.
3290 REM because of the terse error statement.
3300 REM variables used:  external:  INUM$, INUM, INUMX, INUMN;
3310 REM internal:  I, ASCI, LTR$, EPN, DEC, PLS, NGS
3320 DEC=0
3330 EPN=0
3340 FOR IK=1 TO LEN(INUM$)
3350 LTR$=MID$(INUM$,IK,1)
3360 ASCI=ASC(LTR$)
3370 IF ASCI<48 THEN 3400
3380 IF ASCI>57 THEN 3440
3390 GOTO 3580
3400 IF ASCI=43 THEN 3470
3410 IF ASCI=45 THEN 3500
3420 IF ASCI=46 THEN 3560
3430 GOTO 3600
3440 IF ASCI=69 THEN 3530
3450 IF ASCI=101 THEN 3530
3460 GOTO 3600
3470 PLS=PLS+1
3480 IF PLS>2 THEN 3600
3490 GOTO 3580
3500 NGS=NGS+1
3510 IF NGS>2 THEN 3600
3520 GOTO 3580
3530 EPN=EPN+1
3540 IF EPN>1 THEN 3600
3550 GOTO 3580
3560 DEC=DEC+1
3570 IF DEC>1 THEN 3600
3580 NEXT IK
3590 GOTO 3660
3600 PRINT:PRINT TAB(10) STRING$(23,42)+"INPUT ERROR"+STRING$(23,42)
```

```
3610 PRINT TAB(10) "The number you type may include only the numerals 0-9 and"
3620 PRINT TAB(10) "as many as one each of:  . e or E and two each of:  + -"
3630 PRINT TAB(32) "...try again:"
3640 PRINT TAB(10) STRING$(57,42)
3650 GOTO 3760
3660 INUM=VAL(INUM$)
3670 IF EPN=0 THEN 3690
3680 IF INUM=0 THEN 3600
3690 IF INUM>INUMX THEN 3720
3700 IF INUM<INUMN THEN 3720
3710 GOTO 3770
3720 PRINT:PRINT TAB(10) STRING$(23,42)+"INPUT ERROR"+STRING$(23,42)
3730 PRINT TAB(10) "The number you type should be between: " INUMN "and" INUMX
3740 PRINT TAB(32) "...try again:"
3750 PRINT TAB(10) STRING$(57,42)
3760 INUM$="E"
3770 RETURN
3780 REM disk input/output routine
3790 ON ERROR GOTO 3900
3800 INPUT "Which disk drive to use (A or B)";D$
3810 IF D$="a" THEN D$="A":GOTO 3850
3820 IF D$="b" THEN D$="B":GOTO 3850
3830 IF D$="A" OR D$="B" GOTO 3850
3840 PRINT:PRINT:GOTO 3790
3850 A$="":B$="":INPUT "What is the filename to be used";A$
3860 B$=LEFT$(A$,2)
3870 IF B$="A:" OR B$="B:" GOTO 3890
3880 A$=D$+":"+A$
3890 CLOSE:OPEN M$,1,A$:RETURN
3900 IF ERR=53 THEN RESUME 3850
3910 RETURN
```

ACKNOWLEDGEMENTS

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.